Package ‘equivalence’

July 22, 2025

Type Package

Title Provides Tests and Graphics for Assessing Tests of Equivalence

Version 0.8.2

Date 2025-04-27

Maintainer Andrew Robinson <`apro@unimelb.edu.au`>

Depends R (>= 2.10), lattice, boot, PairedData

Imports grid

Description Provides statistical tests and graphics for assessing tests of equivalence. Such tests have similarity as the alternative hypothesis instead of the null. Sample data sets are included.

License GPL-2

NeedsCompilation yes

Author Andrew Robinson [aut, cre]

Repository CRAN

Date/Publication 2025-04-27 23:20:06 UTC

Contents

`equiv.boot` *Regression-based TOST using bootstrap*

Description

This function wraps the regression-based TOST equivalence test inside a bootstrap, extracts and reports the useful quantities, and reports the outcome of the test. The function was written for validating models, and requires paired data points. To use it for this purpose, pass the model predictions as the predictor variable, and the observations (which the predictions are intended to match) as the response variable.

Usage

```
equiv.boot(x, y, alpha = 0.05, b0.ii = 0.25, b1.ii = 0.25, reps = 100,
           b0.ii.absolute = FALSE)
```

Arguments

`x`	the predictor variable (commonly predictions)
`y`	the response variable (commonly observations)
`alpha`	the size of the test
`b0.ii`	the half-length of the region of similarity for the intercept, expressed as a proportion of the estimate or in the same units as the estimate (see b0.ii.absolute).
`b1.ii`	the half-length of the region of similarity for the slope, expressed as a proportion of the estimate.
`reps`	the number of bootstrap replicates required
`b0.ii.absolute`	option to express b0.ii in the same units as the estimate of the intercept.

Details

In each case, if the two one-sided confidence interval is inside the region of similarity then the null hypothesis of dissimilarity is *rejected*.

Value

A list of length 10, comprising

`n`	The effective (non-missing) sample size
`ci.b0`	The intercept TOST confidence interval
`rs.b0`	The intercept region of similarity
`q.b0`	The proportions of simulations below, within, and above, the intercept region of similarity
`Test.b0`	The outcome of the test of the null hypothesis of dissimilarity for the intercept (Reject/Not Reject)

`ci.b1`	The slope TOST confidence interval
`rs.b1`	The slope region of similarity
`q.b1`	The proportions of simulations below, within, and above, the slope region of similarity
`Test.b1`	The outcome of the test of the null hypothesis of dissimilarity for the slope (Reject/Not Reject)
`eff.alpha`	The corrected alpha for each of the two independent tests.

Acknowledgements

Feedback from Mohammad Al-Ahmadi has been very useful for this function.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913.

See Also

`lm`, `boot`, `tost`

Examples

```
# Approximately reproduces the first row from Table 2 of Robinson et al. (2005)

data(pref.4PG)
equiv.boot(pref.4PG$volinc4PG, pref.4PG$stemvolinc)
```

`equiv.p` *Inverts the regression-based TOST equivalence test*

Description

This function generates the TOST intervals for the intercept and the slope of the regression of y on x, and determines the smallest region of indifference in each case that would reject the null hypothesis of dissimilarity.

Usage

```
equiv.p(x, y, alpha = 0.05)
```

Arguments

`x`	The predictor variable - perhaps the model predictions
`y`	The response variable - perhaps the observations
`alpha`	The size of the test

Details

The generated confidence intervals are corrected for experiment-level size of alpha using Bonferroni.

Value

A list of two items:

`Intercept`	The smallest half-length of the interval that leads to rejection of the null hypothesis of dissimilarity for the intercept, in the units of y.
`Slope`	The smallest half-length of the interval that leads to rejection of the null hypothesis of dissimilarity for the slope, in the units of the slope.

Note

The accuracy of the output of this function is contingent on the usual regression assumptions, which are not checked here. Caveat emptor!

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913.

See Also

`tost.data`

Examples

```
data(ufc)
equiv.p(ufc$Height.m.p, ufc$Height.m)
```

`equivalence` *Equivalence Tests*

Description

This package provides tools to perform several equivalence tests.

Note

Recent changes: the `tost.data` function is deprecated as of version 0.5.0; please use `tost`, which provides more functionality.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Schuirmann, D.L. 1981. On hypothesis testing to determine if the mean of a normal distribution is contained in a known interval. Biometrics 37 617.

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

Westlake, W.J. 1981. Response to T.B.L. Kirkwood: bioequivalence testing - a need to rethink. Biometrics 37, 589-594.

Examples

```
data(ufc)

### Tost

tost(ufc$Height.m.p, ufc$Height.m, epsilon = 1)

### equivalence plot

ufc.ht <- ufc[!is.na(ufc$Height),]
equivalence.xyplot(ufc.ht$Height.m ~ ufc.ht$Height.m.p,
                   alpha=0.05, b0.ii=0.1, b1.ii=0.2,
                   xlab="Predicted height (m)",
                   ylab="Measured height (m)")
```

---

equivalence-deprecated

Deprecated Functions in Equivalence package

Description

These functions are provided for compatibility with older versions of R only, and may be defunct as soon as the next release.

Usage

```
tost.data(x, null = 0, alpha = 0.05, Epsilon = 0.36, absolute = FALSE)
```

Arguments

`x`	the sample of paired differences
`null`	the value of the parameter in the null hypothesis
`alpha`	test size
`Epsilon`	magnitude of region of similarity
`absolute`	choose units: absolute (TRUE) or relative to the standard deviation (FALSE).

Details

The original help page for these functions is available at `help("oldName-deprecated")` (note the quotes). Functions in packages other than the base package are listed in `help("pkg-deprecated")`.

`tost.data` is superceded by `tost`.

See Also

`Deprecated`

---

equivalence.xyplot

Constructs graphical regression-based tests of equivalence inside a lattice coplot

---

Description

Implements regression-based tests of equivalence within lattice graphics.

Usage

```
 equivalence.xyplot(formula, alpha, b0.ii, b1.ii,
add.smooth=FALSE, b0.absolute=FALSE, ...)
```

Arguments

`formula`	a formula describing the form of conditioning plot. See the manual entry for `xyplot` for more details.
`alpha`	the size of the test
`b0.ii`	the half-length of the region of similarity for the intercept, can be relative or absolute (see below).
`b1.ii`	the half-length of the region of similarity for the slope.
`add.smooth`	adds a loess smooth to the graph.
`b0.absolute`	is b0.ii in absolute or relative units?
`...`	extra arguments passed on to `xyplot`

Details

The graphic created by this function was proposed by Robinson et al. (2005) as a visual summary of the regression-based TOST. At first glance the graph will look messy; interpretation eases with practice. The following points should be noted.

LS line: A black, solid line of best fit is added.

Mean: A grey vertical bar indicates the mean of x and the TOST confidence interval for the intercept.

b0 R.S.: A shaded polygon is the region of similarity of the intercept, to test the model bias.

Test b0: If the grey vertical bar is within the shaded polygon then reject the null hypothesis of dissimilarity. This is a test of *bias*.

...1 - If the *region* is too low then the predictions are too low.

...2 - If the *region* is too high then the predictions are too high.

...3 - If the *region* is too narrow then the predictions are too variable.

b1 C.I.: A black vertical bar undermeath the grey bar represents a confidence interval for the slope of the line of best fit.

b1 R.S.: Two black dashed lines are added representing the region of similarity.

Test b1: If the black bar is within the angle described by the dashed black lines then the slope of the observed/predicted regression is significantly similar to 1. This is a test of *proportionality*.

...1 - If the *bar* is too high then the slope is too high; the model over-predicts the higher observations and under-predicts the lower observations.

...2 - If the *bar* is too low then the slope is too low; the model underpredicts the higher observations and overpredicts the lower observations (analogous to regression to the mean).

...3 - If the *bar* is too narrow then the predictions are too variable.

The implementation in Robinson et al. (2005) required shifting so that the predictor has 0 mean. This hack has been removed here so that the basic graph object is a plot of the two variables being compared.

Value

Run for its side effect of producing a lattice plot object.

Warning

The accuracy of the output of this function is contingent on the usual regression assumptions, which are not checked here. *Caveat emptor*! Consider using equiv.boot() for a bootstrap-based solution. Transforming either variable will probably complicate the analysis considerably.

Acknowledgements

Feedback from Mohammad Al-Ahmadi has been very useful for this function.

Note

This version produces a regression-based TOST for each level of the conditioning factor. There may be an argument for pooling the test across these levels, in which case some prepanel computations will be helpful.

The TOST requires only estimates and standard errors from the data. Therefore the linear model used in the panel function can be replaced by any model that will produce suitable estimates. For example, in applying this function to hierarchical data we have had success using lme() instead.

I'm not entirely convinced that all these lines on one image are a good idea. It's straightforward to remove some, or change the colours. Recommendations for graphics that are visually cleaner are welcome.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913.

See Also

`tost.stat`, `xyplot`, `equiv.boot`

Examples

```
data(pref.4PG)
equivalence.xyplot(pref.4PG$stemvolinc ~ pref.4PG$volinc4PG,
                   alpha=0.05, b0.ii=0.25, b1.ii=0.25, add.smooth=TRUE,
                   xlab=expression(paste("4PG decadal volume growth (", m^3,
                       ha^-1, decade^-1, ")", sep="")),
                   ylab=expression(paste("Measured decadal volume growth (",
                       m^3, ha^-1, decade^-1, ")", sep="")))

data(pref.LAI)
equivalence.xyplot(pref.LAI$lai.pa ~ pref.LAI$lai.bl,
                   alpha=0.05, b0.ii=0.25, b1.ii=0.25,
                   xlab=expression(paste("LAI Beer-Lambert (", m^2, m^-2, ")",
                       sep="")),
                   ylab=expression(paste("LAI Ceptometer (", m^2, m^-2, ")",
```

```
                         sep="")))


data(ufc)
ufc.ht <- ufc[!is.na(ufc$Height),]
equivalence.xyplot(ufc.ht$Height.m ~ ufc.ht$Height.m.p,
                   alpha=0.05, b0.ii=0.1, b1.ii=0.2,
                   xlab="Predicted height (m)",
                   ylab="Measured height (m)")

equivalence.xyplot(ufc.ht$Height.m ~ ufc.ht$Height.m.p | ufc.ht$Species,
                   alpha=0.05, b0.ii=0.1, b1.ii=0.2,
                   xlab="Predicted height (m)",
                   ylab="Measured height (m)",
                   subset=ufc.ht$Species %in%
                        levels(ufc.ht$Species)[table(ufc.ht$Species)>5])
```

`pref.4PG` *Measured and simulated data from PREF, northern Idaho, USA, and 4-PG*

Description

These data are the juxtaposition of model output and field measurements for the Priest River Experimental Forest in northern Idaho, USA. The model was a process-based aspatial forest stand model called 4-PG, developed from 3-PG by Duursma (2004). The data were used to demonstrate a regression-based TOST in Robinson et al (2005).

Usage

```
data(pref.4PG)
```

Format

A dataset with 35 observations on 33 variables, of which the following two were used for the model validation exercise.

`stemvolinc`	measured decadal stem growth in m^3/ha
`volinc4PG`	predicted decadal stem growth in m^3/ha

Details

Nothing in particular. Devil's club (*Oplopanax horridus*) is very painful.

Source

The data are documented in Duursma (2004) and Robinson et al (2005).

References

Duursma, R.A. 2004. A simple process-based model of forest growth, and a test for the Priest River Experimental Forest. Ph.D. Thesis, University of Idaho, 169 p.

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913

Examples

```
data(pref.4PG)
```

`pref.LAI` *Measured Leaf Area Index data from PREF, northern Idaho, USA*

Description

These data are the juxtaposition of model output and field measurements for 36 forest plots in the Priest River Experimental Forest in northern Idaho, USA. The model was based on the Beer-Lambert model (Duursma et al. 2003). The data were used to demonstrate a regression-based TOST in Robinson et al (2005).

Usage

```
data(pref.LAI)
```

Format

A dataset with 36 observations on 8 variables, the following two of which were used for the test of equivalence:

`lai.pa` LAI as estimated using allometric functions applied to measurements of trees within the sample plot.

`lai.bl` LAI as estimated using the Beer-Lambert model from measurements of interception using a hand-held ceptometer.

Details

Nothing in particular. PREF, and northern Idaho, are very attractive.

Source

The data are documented in Duursma et al. (2003) and Robinson et al (2005).

References

Duursma, R.A., J.D. Marshall, and A.P. Robinson. 2003. Leaf area index inferred from solar beam transmission in mixed conifer forests on complex terrain. Agricultural and Forest Meteorology 118, 221-236.

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913.

Examples

```
data(pref.LAI)
```

`print.tost` *Print methods for TOST objects*

Description

Printing objects of class '"tost"' by simple 'print' methods.

Usage

```
  ## S3 method for class 'tost'
print(x, ...)
```

Arguments

`x`	object of class '"tost"'
`...`	arguments to be passed to other functions.

Details

The function inherits infrastructure from R's print.htest, so a number of elements have been copied from the help file of that function.

Value

the argument 'x', invisibly, as for all 'print' methods.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

See Also

`tost`

Examples

```
data(ufc)
tost(ufc$Height.m.p, ufc$Height.m, epsilon = 1, paired = TRUE)
```

`ptte.data` *Computes a paired t-test for equivalence from a single sample of a normally-distributed population*

Description

This function computes the test and key test quantities for the paired t-test for equivalence, as documented in Wellek (2003, pp 77-80). This function computes the test from a sample of a normally-distributed population.

Usage

```
ptte.data(x, alpha = 0.05, Epsilon = 0.25)
```

Arguments

`x`	paired differences
`alpha`	test size
`Epsilon`	magnitude of region of similarity

Details

This test requires the assumption of normality of the population. Under that assumption the test is the uniformly most powerful invariant test (Wellek, 2003, pp. 78-79).

The function as documented by Wellek (2003) uses units relative to the standard deviation, noting (p. 12) that 0.25 corresponds to a strict test and 0.5 to a liberal test.

Value

A list with the following components

`Dissimilarity`	the outcome of the test of the null hypothesis of dissimilarity
`Mean`	the mean of the sample
`StdDev`	the standard deviation of the sample
`n`	the sample size
`alpha`	the size of the test
`missing`	the number of observations missing
`Epsilon`	the magnitude of the region of similarity
`cutoff`	the critical value
`Tstat`	the test statistic; if Tstat < cutoff then the null hypothesis is rejected.
`Power`	the power of the test evaluated at the observed value

Note

This test requires the assumption of normality of the population. Under that assumption the test is the uniformly most powerful invariant test (Wellek, 2003, pp. 78-79). The exposition in Robinson and Froese (2004) mistakenly omits the square root of the F-quantile.

Author(s)

Andrew Robinson<`A.Robinson@ms.unimelb.edu.au`>

References

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

See Also

`ptte.stat`, `tost.data`

Examples

```
data(ufc)
ptte.data(ufc$Height.m.p - ufc$Height.m)
```

`ptte.stat` *Computes a paired t-test for equivalence from the mean and standard deviation of a sample from a normally-distributed population*

Description

This function computes the test and key test quantities for the paired t-test for equivalence, as documented in Wellek (2003, pp 77-80). This function computes the test from the mean and standard deviation of a sample of paired differences from a normally-distributed population.

Usage

```
ptte.stat(mean, std, n, alpha = 0.05, Epsilon = 0.25)
```

Arguments

`mean`	the sample mean
`std`	the sample standard deviation
`n`	sample size
`alpha`	test size
`Epsilon`	magnitude of region of similarity

Details

This test requires the assumption of normality of the population. Under that assumption the test is the uniformly most powerful invariant test (Wellek, 2003, pp. 78-79). This version of the test can be applied post-hoc to any testing situation in which you have the mean, standard deviation, and sample size, and are confident that the sample is drawn from a normally-distributed population.

The function as documented by Wellek (2003) uses units relative to the standard deviation, noting (p. 12) that 0.25 corresponds to a strict test and 0.5 to a liberal test.

Value

A list with the following components

`Dissimilarity`	the outcome of the test of the null hypothesis of dissimilarity
`Mean`	the mean of the sample
`StdDev`	the standard deviation of the sample
`n`	the sample size
`alpha`	the size of the test
`Epsilon`	the magnitude of the region of similarity
`cutoff`	the critical value
`Tstat`	the test statistic; if Tstat < cutoff then the null hypothesis is rejected.
`Power`	the power of the test evaluated at the observed value

Note

The exposition in Robinson and Froese (2004) mistakenly omits the square root of the F-quantile.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

See Also

`ptte.data`, `tost.stat`

Examples

```
data(ufc)
ptte.stat(mean(ufc$Height.m.p - ufc$Height.m, na.rm=TRUE),
  sd(ufc$Height.m.p - ufc$Height.m, na.rm=TRUE),
  sum(!is.na(ufc$Height.m.p - ufc$Height.m)))
```

`rtost` *Computes a robust TOST for equivalence from paired or unpaired data*

Description

This function computes the TOST and key TOST quantities for the two one-sided test for equivalence (Schuirmann 1981; Westlake 1981), using the robust t-test of Yuen (Yuen and Dixon 1973, Yuen 1974) in place of the standard Welch t test (t.test in stats package). The Yuen t test makes no assumption of normality. The function computes the robust TOST for a sample of paired differences or for two samples. The function performs almost as well as the Welch t test when the population distribution is normal and is more robust than the Welch t test in the face of non-normality (e.g., distributions that are long-tailed, heteroscedastic, or contaminated by outliers; Yuen and Dixon 1973, Yuen 1974).

Usage

```
rtost(x, y = NULL, alpha = 0.05, epsilon = 0.31, tr = 0.2,  ...)
```

Arguments

`x`	the first (or only) sample
`y`	the second sample
`alpha`	test size
`tr`	the proportion (percent/100) of the data set to be trimmed
`epsilon`	magnitude of region of similarity
`...`	arguments to be passed to yuen.t.test

Details

The rtost function is wrapped around the yuen t test from the PairedData package, a robust variant of the t test using trimmed means and winsorized variances. It provides tosts for the same range of designs, accepts the same arguments, and handles missing values the same way as tost. For the tost, the user must set epsilon, which is the magnitude of region similarity. Warning: with tr > 0.25 type I, error control might be poor.

Value

A list with the following components

`mean.diff`	the mean of the difference
`se.diff`	the standard error of the difference
`alpha`	the size of the test
`ci.diff`	the 1-alpha confidence interval for the difference
`df`	the degrees of freedom used for the confidence interval
`epsilon`	the magnitude of the region of similarity

`result`	the outcome of the test of the null hypothesis of dissimilarity
`p.value`	the p-value of the significance test
`check.me`	the confidence interval corresponding to the p-value

Note

This test is the tost for equivalence wrapped around the robust, trimmed mean, winsorized variance yuen.t.test (yuen in PairedData).

Author(s)

Gregory Belenky `<belenky@wsu.edu>`

References

Schuirmann, D.L. 1981. On hypothesis testing to determine if the mean of a normal distribution is contained in a known interval. Biometrics 37, 617.

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

Westlake, W.J. 1981. Response to T.B.L. Kirkwood: bioequivalence testing - a need to rethink. Biometrics 37, 589-594.

Yuen, K.K. (1974) The two-sample trimmed t for unequal population variances. Biometrika, 61, 165-170.

Yuen, K.K., and Dixon, W.J. (1973) The approximate behavior and performance of the two-sample trimmed t. Biometrika, 60, 369-374.

See Also

`tost`, `yuen.t.test`

Examples

```
data(ufc)
rtost(ufc$Height.m.p, ufc$Height.m, epsilon = 1, tr = 0.2)
```

`tost` *Computes a TOST for equivalence from paired or unpaired data*

Description

This function computes the test and key test quantities for the two one-sided test for equivalence, as documented in Schuirmann (1981) and Westlake (1981). The function computes the test for a sample of paired differences or two samples, assumed to be from a normally-distributed population.

Much code in the function has been copied and adapted from R's t.test.default function.

Usage

```
tost(x, y = NULL, epsilon = 1, paired = FALSE, var.equal = FALSE,
              conf.level = 0.95, alpha = NULL,
    ...)
```

Arguments

`x`	the first (or only) sample
`y`	the second sample
`epsilon`	magnitude of region of similarity
`paired`	a logical indicating whether you want a paired tost
`var.equal`	a logical variable indicating whether to treat the two variances as being equal. If 'TRUE' then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
`conf.level`	confidence level of the interval
`alpha`	test size (for backwards-compatibility, overrides conf.level)
`...`	arguments to be passed to other functions.

Details

The function inherits infrastructure from R's t.test.default, so a number of elements have been copied from the help file of that function.

This test requires the assumption of normality of the population, or an invocation of large-sample theory. The function wraps around the `t.test` function, so it provides tosts for the same range of designs, accepts the same arguments, and handles missing values the same way.

If 'paired' is 'TRUE' then both 'x' and 'y' must be specified and they must be the same length. Missing values are silently removed (in pairs if 'paired' is 'TRUE'). If 'var.equal' is 'TRUE' then the pooled estimate of the variance is used. By default, if 'var.equal' is 'FALSE' then the variance is estimated separately for both groups and the Welch modification to the degrees of freedom is used.

Value

A list with the following components

`estimate`	the mean of the difference
`se.diff`	the standard error of the difference
`alpha`	the size of the test
`data.name`	a character string giving the name(s) of the data
`ci.diff`	the 1-alpha confidence interval for the difference
`parameter`	the degrees of freedom used for the confidence interval
`epsilon`	the magnitude of the region of similarity
`result`	the outcome of the test of the null hypothesis of dissimilarity
`method`	a character string indicating what type of t-test was performed

`null.value`	the specified hypothesized value of the mean or mean difference depending on whether it was a one-sample tost or a two-sample tost.
`tost.p.value`	the p-value of the tost significance test
`tost.interval`	the two one-sided confidence interval corresponding to the test.

Note

This test requires the assumption of normality of the population. The components of the test are t-based confidence intervals, so the Central Limit Theorem and Slutsky's Theorem may be relevant to its application in large samples.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Schuirmann, D.L. 1981. On hypothesis testing to determine if the mean of a normal distribution is contained in a known interval. Biometrics 37 617.

Robinson, A.P., and R.E. Froese. 2004. Model validation using equivalence tests. Ecological Modelling 176, 349–358.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

Westlake, W.J. 1981. Response to T.B.L. Kirkwood: bioequivalence testing - a need to rethink. Biometrics 37, 589-594.

See Also

`tost.stat`, `ptte.data`

Examples

```
data(ufc)
tost(ufc$Height.m.p, ufc$Height.m, epsilon = 1, paired = TRUE)
```

`tost.stat` *Computes a TOST for equivalence from sample statistics*

Description

This function computes the test and key test quantities for the two one-sided test for equivalence, as documented in Schuirmann (1981) and Westlake (1981). This function computes the test from the statistics of a sample of paired differences of a normally-distributed population.

Usage

```
tost.stat(mean, std, n, null = 0, alpha = 0.05, Epsilon = 0.36)
```

Arguments

`mean`	sample mean
`std`	sample standard deviation
`n`	sample size
`null`	the value of the parameter in the null hypothesis
`alpha`	test size
`Epsilon`	magnitude of region of similarity

Details

This test requires the assumption of normality of the population.

Value

A list with the following components

`Dissimilarity`	the outcome of the test of the null hypothesis of dissimilarity
`Mean`	the mean of the sample
`StdDev`	the standard deviation of the sample
`n`	the non-missing sample size
`alpha`	the size of the test
`Epsilon`	the magnitude of the region of similarity
`Interval`	the half-length of the two one-sided interval

Note

This test requires the assumption of normality of the population. The components of the test are t-based confidence intervals, so the Central Limit Theorem and Slutsky's Theorem may be relevant to its application in large samples.

Author(s)

Andrew Robinson `<A.Robinson@ms.unimelb.edu.au>`

References

Schuirmann, D.L. 1981. On hypothesis testing to determine if the mean of a normal distribution is contained in a known interval. Biometrics 37 617.

Wellek, S. 2003. Testing statistical hypotheses of equivalence. Chapman and Hall/CRC. 284 pp.

Westlake, W.J. 1981. Response to T.B.L. Kirkwood: bioequivalence testing - a need to rethink. Biometrics 37, 589-594.

See Also

`tost.data`, `ptte.stat`

Examples

```
data(ufc)
tost.stat(mean(ufc$Height.m.p - ufc$Height.m, na.rm=TRUE),
  sd(ufc$Height.m.p - ufc$Height.m, na.rm=TRUE),
  sum(!is.na(ufc$Height.m.p - ufc$Height.m)))
```

`ufc` *Upper Flat Creek cruise data*

Description

These are forest measurement data from the Upper Flat Creek unit of the University of Idaho Experimental Forest, measured in 1991. The inventory was based on variable radius plots with 6.43 sq. m. per ha. BAF (Basal Area Factor). The forest stand was 121.5 ha.

Usage

```
data(ufc)
```

Format

A data frame with 633 observations on the following 12 variables.

Plot plot label

Tree tree label

Species species kbd

Dbh tree diameter at 1.37 m. from the ground, measured in millimetres.

Height tree height measured in decimetres

Height.ft tree height converted to feet

Dbh.in tree diameter converted to inches

Height.ft.p predicted tree height in feet

Height.m tree height in metres

Height.m.p predicted tree height in metres

Details

Plots that were measured with no trees are signified in the dataset by lines that have blank species codes and missing DBH.

Source

The data are provided courtesy of Harold Osborne and Ross Appelgren of the University of Idaho Experimental Forest. The predicted height comes from the height/diameter model documented in Wykoff et al. (1982). The data and model were used in Robinson et al. (2005).

References

Robinson, A.P., R.A. Duursma, and J.D. Marshall. 2005. A regression-based equivalence test for model validation: shifting the burden of proof. Tree Physiology 25, 903-913.

Wykoff,W.R., N.L. Crookston, and A.R. Stage. 1982. User's guide to the stand Prognosis model. GTR-INT 133, USDA Forest Service, Intermountain Research Station, Ogden, UT, 113 p.

Examples

```
data(ufc)
```

Index